Table of Contents

Pleiades

Pleiades: Introduction

Pleiades is the primary supercomputer at NAS. Originally installed in 2008 with 51,200 cores, it has been further expanded at various stages. The following articles provide hardware information at varying levels of detail:

- Pleiades Hardware Overview - a high-level overview of the Pleiades system architecture, including resource summaries of the compute and front-end nodes, the interconnect, and the storage capacity.

- Pleiades Configuration Details - focuses on the hardware hierarchy (from the processors to the whole cluster) and provides more detailed configuration statistics on the processors and their associated memory.

- Harpertown Processors, Nehalem-EP Processors, and Westmere Processors (3 articles) - provide configuration diagrams and additional information such as core labeling, instruction set, hyperthreading, and Turbo Boost, for each of Pleiades' three processor types.

- Comparison among Harpertown, Nehalem-EP, and Westmere - points out the differences and similarities among the three processor types.

- Pleiades Home Filesystem - information on quota and backup policies on the home filesystem.

- Pleiades Lustre Filesystems - details the configurations of the Lustre filesystems and users' quotas on these filesystems.

- Pleiades Interconnect - information on the topology, latency, and bandwidth of the Pleiades InfiniBand fabric.

- Pleiades Front-End Usage Guidelines - guidelines on using the front-end nodes and bridge nodes.

Pleiades Hardware Overview

Pleiades, the seventh most powerful supercomputer in the world, represents NASA's state-of-the-art technology for meeting the agency's supercomputing requirements, enabling NASA scientists and engineers to conduct modeling and simulation for NASA missions. This distributed-memory SGI ICE cluster is connected with InfiniBand in a dual-plan hypercube technology.

This system contains the following types of Intel Xeon processors: X5670 (Westmere), X5570 (Nehalem), E5472 (Harpertown) and NVIDIA M2090 GPU. Pleiades is named after the astronomical open star cluster of the same name.

System Architecture

- Manufacturer - SGI
- 185 racks (11,776 nodes)
- 1.34 Pflop/s peak cluster
- 1.09 Pflop/s LINPACK rating (June 2011, using 11,648 nodes)
- 2 racks (64 nodes total) enhanced with NVIDIA graphics processing unit (GPU): 52 teraflops total
- Total cores: 112,896 (32,768 additional GPU cores)
- Total memory: 191 TB
- Nodes
    - 4,608 Westmere nodes
        - 2 six-core processors per node
        - 4,480 nodes of Xeon X5670 and 128 nodes of Xeon X5675 (Westmere) processors
        - Processor speed: 2.93 GHz (X5670) or 3.06 GHz (X5675)
        - Cache: 12 MB Intel Smart Cache for 6 cores
        - Memory Type: DDR3 FB-DIMMs
        - 2 GB per core, 24 GB per node
        - InfiniBand® QDR host channel adapter
    - 1,280 Nehalem nodes
        - 2 quad-core processors per node
        - Xeon X5570 (Nehalem) processors
        - Processor speed: 2.93 GHz
        - Cache: 8 MB Intel Smart Cache for 4 cores
        - Memory Type: DDR3 FB-DIMMs
        - 3 GB per core, 24 GB per node
        - InfiniBand® DDR host channel adapter
    - 5,824 Harpertown nodes
        - 2 quad-core processors per node
        - Xeon E5472 (Harpertown) processors
        - Processor speed: 3 GHz
        - Cache: 6 MB per pair of cores
        - Memory Type: DDR2 FB-DIMMs

        - ◊ 1 GB per core, 8 GB per node (Except 64 nodes with 2GB per core)
        - ◊ InfiniBand® DDR host channel adapter
    - ♦ 64 GPU/Westmere nodes
        - ◊ 2 six-core Westmere processors per node
            - · Xeon X5670 processors
            - · Processor speed: 2.93 GHz
            - · Cache: 12 MB Intel Smart Cache for 6 cores
            - · Memory Type: DDR3 FB-DIMMs
            - · 4 GB per core, 48 GB per node
            - · InfiniBand® QDR host channel adapter
        - ◊ 1 NVIDIA Tesla M2090 GPU (512 CUDA cores) per node
            - · Processor speed: 1.3 GHz
            - · Memory size: 6 GB per node

Subsystems

- 14 front-end nodes
    - ◊ 2 quad-core processors per node
    - ◊ Xeon E5472 (Harpertown) processors
    - ◊ Processor speed - 3GHz
    - ◊ 16 GB per node
    - ◊ 1 Gigabit Ethernet connection
- 2 bridge nodes
    - ◊ 2 quad-core processors per node
    - ◊ Xeon E5472 (Harpertown) processors
    - ◊ Processor speed - 3GHz
    - ◊ 64 GB per node
    - ◊ 10 Gigabit Ethernet connection
- 1 PBS server
    - ◊ 2 quad-core processors per node
    - ◊ Xeon E5472 (Harpertown) processors
    - ◊ Processor speed - 3GHz
    - ◊ 16 GB per node

Interconnects

- Internode - InfiniBand, with all nodes connected in a partial 11D hypercube
- Two independent InfiniBand fabrics (ib0, ib1)
- Infiniband DDR, QDR
- Gigabit Ethernet management network

Storage

- SGI® InfiniteStorage NEXIS 9000 home filesystem
- 12 DDN 9900 RAIDs - 6.9 PB total
- 7 Oracle Lustre cluster-wide filesystems, each containing:

- 1 Metadata server (MDS)
- 8 Object Storage Servers (OSS)
- 60 - 120 Object Storage Targets (OST)

Operating Environment

- Operating system - SUSE® Linux®
- Job Scheduler - PBS®
- Compilers - Intel and GNU C, C++ and Fortran
- MPI - SGI MPT, MVAPICH2, Intel MPI

Related Links

Links related to the Pleiades system.

- Pleiades Configuration Details
- Pleiades Front-End Usage Guidelines

Pleiades Configuration Details

DRAFT

This article is being reviewed for completeness and technical accuracy.

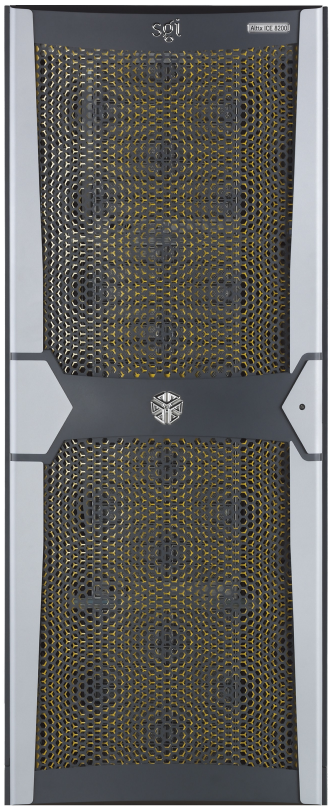
Pleiades Hardware Hierarchy

The hardware hierarchy from a single processor to the whole Pleiades supercluster is described below:

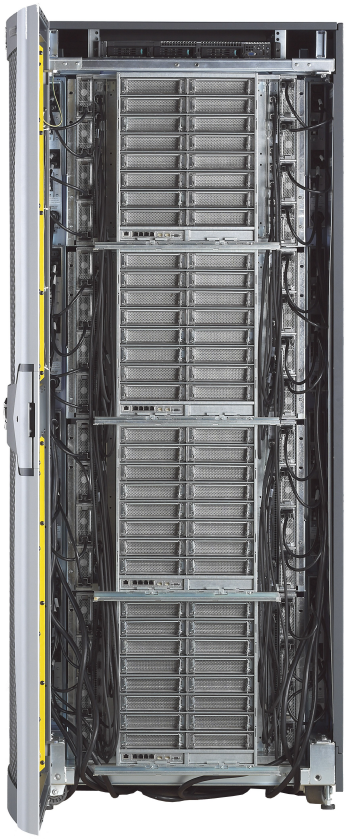
- one quad-core (Harpertown, Nehalem-EP) or 6-core (Westmere) processor per socket
- 2 sockets in 1 node
- 16 compute nodes (labeled as n0 - n15) in 1 IRU (individual rack units)
- 4 IRUs (labeled as i0 - i3) in 1 rack
- 91 Harpertown racks (labeled as r1-r91);
  20 Nehalem-EP racks (labeled as r161-170, r177-186):
  and 73 Westmere racks (labeled as r129-160, r171-176, r187-218, r219, r221-r222)
  in the Pleiades supercluster

The nomenclature of a Pleiades compute node is based on which rack and IRU it is on. For example, r1i0n15 is the node 15 in IRU 0 of rack 1.

Below are the front views of a typical rack:



Front view of a rack



Front-open view of a rack (includes 4 IRUs)

Processor and Memory Subsystems Statistics

Below are detailed configuration statistics for the processor and memory subsystems for all Pleiades nodes:

	Pleiades Processor and Memory Subsustems Statistics				
Hostname	**pfe[1-12] bridge[1-2]**	**pbspl1, pbspl3**	**r[1-91]i[0-3]n[0-15]**	**r[161-170,177-186] i[0-3]n[0-15]**	**r[129-160,1 219, 221-22 i[0-3]n[0-15**
Function	front-end * bridge node with Columbia CXFS filesystems mounted	PBS server	compute	compute	compute
Architecture	ICE 8200EX	ICE 8200EX	ICE 8200EX	ICE 8200EX	ICE 8400EX
			Processor		
CPU	**Quad-Core Xeon E5472 (Harpertown)**	**Quad-Core Xeon E5472 (Harpertown)**	**Quad-Core Xeon E5472 (Harpertown)**	**Quad-Core Xeon X5570 (Nehalem-EP)**	**6-Core Xeon X567 (r221-222)**

					(Westmere
CPU-Clock	**3.00 GHz**	**3.00 GHz**	**3.00 GHz**	**2.93 GHz**	**2.93/3.06 (r**
Floating Point Operations per cycle per Core	4	4	4	4	4
# of Cores/blade (or node)	8	8	8	8	12
Total # of nodes	.	.	5,824	1,280	4,672
Total # of Cores	.	.	46,592	10,240	56,064
			Memory		
L1 Cache	Local to each core; Instruction cache: 32K Data cache: 32K; 32B/cycle;	Local to each core; Instruction cache: 32K Data cache: 32K; 32B/cycle;	Local to each core; Instruction cache: 32K Data cache: 32K; 32B/cycle;	Local to each core; Instruction cache: 32K Data cache: 32K; 32B/cycle;	Local to eac Instruction c Data cache 32B/cycle;
L2 Cache	12MB on-die for the Quad-Core; 6MB per core pair; shared by the two cores.	12MB on-die for the Quad-Core; 6MB per core pair; shared by the two cores. L2 Cache speed: 3 GHz	12MB on-die for the Quad-Core; 6MB per core pair; shared by the two cores. L2 Cache speed: 3 GHz	256 KB per core	256 KB per
L3 Cache	N/A	N/A	N/A	8 MB shared by the four cores	12 MB shar cores
TLB	local to each core	local to each core	local to each core	local to each core	local to eac
Default Page Size	4 KB	4 KB	4 KB	4 KB	4 KB
Local Memory/Core	**2 GB (pfe[1-12]); 8 GB (bridge[1-2]; Fully Buffered DDR2 DIMM**	**2 GB**	**1 GB**	**3 GB; DDR3**	**2 GB: DDR**
		16 GB	8 GB	24 GB	24 / 48(r21

Total Memory/node	16 GB (pfe[1-12]); 64 GB (bridge[1-2])				
Front-Side Bus	1600 MHz; 25.6 GB/sec read 12.8 GB/sec write	1600 MHz; 25.6 GB/sec read 12.8 GB/sec write	1600 MHz; 25.6 GB/sec read 12.8 GB/sec write	N/A	N/A
Memory Controller	N/A	N/A	N/A	32 GB/sec read/write	32 GB/sec r
QuickPath Interconnect	N/A	N/A	N/A	25.6 GB/sec	25.6 GB/se

One of the Harpertown racks, rack 32, provides 16 GB of memory per node, double the size of per-node memory available in the other Harpertown racks.

Related articles:

Pleiades Hardware Overview

Harpertown Processors

Nehalem-EP Processors

Westmere Processors

Comparison among Harpertown, Nehalem-EP, and Westmere

Pleiades Home Filesystem

Pleiades Lustre Filesystems

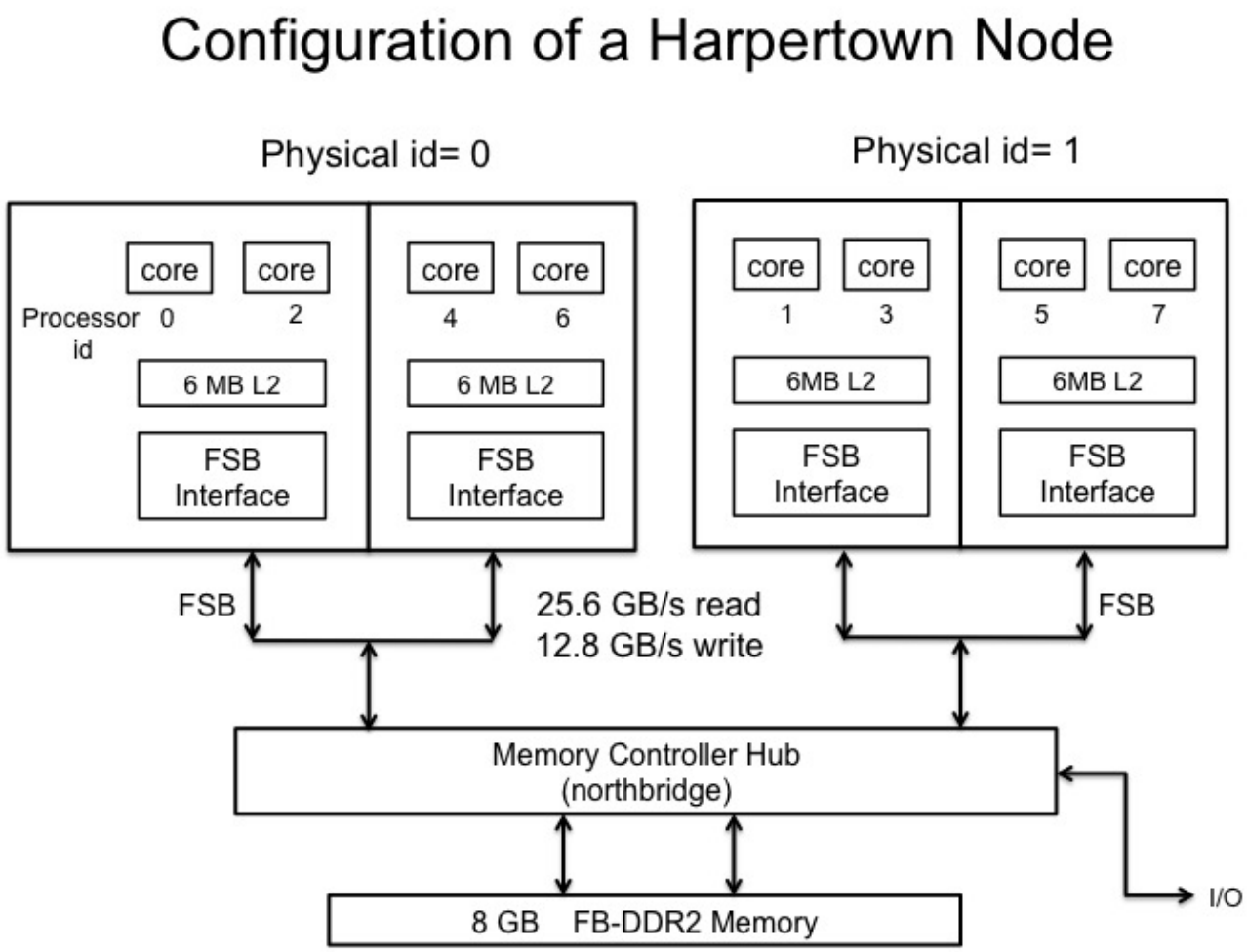
Pleiades Front-End Usage Guidelines

Harpertown Processors

DRAFT

This article is being reviewed for completeness and technical accuracy.

Configuration of a Harpertown node:



Core Labeling:

The core labeling as shown in this diagram is obtained from the command *cat /proc/cpuinfo*. Note that in the first socket (i.e., phyiscal id=0), the four cores are labeled 0, 2, 4, and 6, and are not contiguous. Similarly, in the second socket (physical id=1), they are labeled as 1, 3, 5, and 7. In addition, each core pair (0,2), (4,6), (1,3) and (5,7) shares a 6MB L2 cache.

For performance consideration, care must be taken if one tries to use tools such as *dplace* to pin processes to specific processors. Be aware of the non-contiguous nature of the labeling and the sharing of L2 cache per core pair. Also, when using the SGI MPT library, the environment variable **MPI_DSM_DISTRIBUTE** has been set to *off* for the Harpertown nodes since setting MPI_DSM_DISTRIBUTE to *on* causes the processes to be pinned to processors in a contiguous order. For example, MPI ranks 0-7 are pinned to processors 0-7, respectively. This results in bad performances for most applications.

SSE4 Instruction Set:

Intel's Streaming SIMD Extensions 4.1 (SSE4.1) instruction set is included in the Harpertown processors.

Since the instruction set is upward compatible, an application which is compiled with -xSSE4.1 (with Intel version 11 compiler) can run on either Harpertown or Nehalem-EP or Westmere processors. An application which is compiled with -xSSE4.2 can run ONLY on Nehalem-EP or Westmere processors.

If you wish to have a single executable that will run on any of the three Pleiades processor types with suitable optimization to be determined at run time, you can compile your application with -O3 -ipo -axSSE4.2,xSSE4.1

Hyperthreading:

Not available.

Turbo Boost:

Not available.

Front-Side Bus

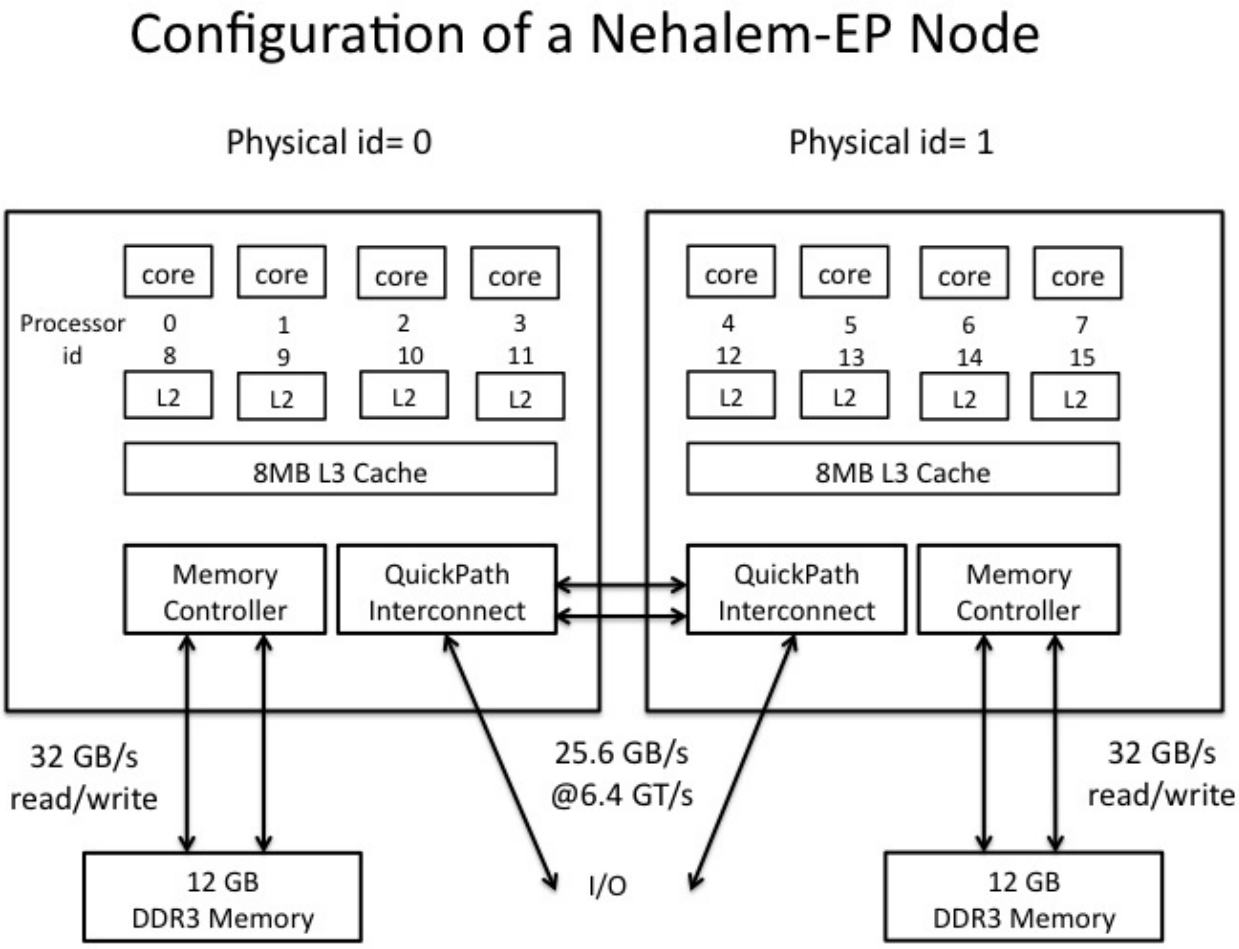
The Harpertown (Quad-Core Intel Xeon Processor E5472) processors at NAS use 1600 MHz Front-Side Bus (FSB). The processor transfers data four times per bus clock (4X data transfer rate, as in AGP 4X). Along with the 4X data bus, the address bus can deliver addresses two times per bus clock and is referred to as a double-clocked or a 2X address bus. In addition, the Request Phase completes in one clock cycle. Working together, the 4X data bus and 2X address bus provide a data bus bandwidth of up to 12.8 GBytes per second. The FSB is also used to deliver interrupts.

Nehalem-EP Processors

DRAFT

This article is being reviewed for completeness and technical accuracy.

Configuration of a Nehalem-EP node:



Core Labeling:

Unlike Harpertown, the core labeling in Nehalem-EP (and also Westmere) is contiguous. That is, cores 0-3 are in first socket and cores 4-7 are in the second socket.

When using the SGI MPT library, the enviroment variable **MPI_DSM_DISTRIBUTE** is set to *on* by default for the Nehalem-EP (and also Westmere) nodes.

SSE4 Instruction Set:

Intel's Streaming SIMD Extensions 4.2 (SSE4.2) instruction set is included in the Nehalem-EP processors.

Since the instruction set is upward compatible, an application that is compiled with -xSSE4.1 (with Intel version 11 compiler) can run on either Harpertown or Nehalem-EP or Westmere processors. An application that is compiled with -xSSE4.2 can run ONLY on Nehalem-EP or Westmere processors.

If you wish to have a single executable that will run on any of the three Pleiades processor types with suitable optimization to be determined at run time, you can compile your application with -O3 -ipo -axSSE4.2,xSSE4.1

Hyperthreading:

On Nehalem-EP (and also Westmere), hyperthreading is available by user request, for example by asking for more than 8 MPI ranks per Nehalem-EP node.

When hyperthreading is requested, the OS views each physical core as two logical processors and can assign two threads to it.

Preliminary benchmarking by NAS shows that many jobs would benefit from using hyperthreading. Therefore, it is currently turned ON, meaning that it is available if a job requests it.

Mapping of Physical Cores and Logical Processor IDs

Physical id	Core id	Processor id Hyperthreading OFF	Processor id Hyperthreading ON
0	0	0	0 ; 8
0	1	1	1 ; 9
0	2	2	2 ; 10
0	3	3	3 ; 11
1	4	4	4 ; 12
1	5	5	5 ; 13
1	6	6	6 ; 14
1	7	7	7 ; 15

With hyperthreading, one can run an MPI code with 16 processes instead of just 8 per Nehalem-EP node. Each of the 16 processes will be assigned to run on one logical processor. In reality, two processes are running on the same physical core. If one process does not keep the functional units in the core busy all the time and can share the resources in the core with another process, then running in this mode will take less than 2 times the walltime compared to running only 1 process on the core. This can improve the overall

throughput as demonstrated in the following example:

Example: Consider the following scenario with a job that uses 16 MPI ranks. Without hyperthreading we would use:

#PBS -lselect=2:ncpus=8:mpiprocs=8 -lplace=scatter:excl

and the job will use 2 nodes with 8 processes per node. Suppose that the job takes 1000 seconds when run this way. If we run the job with hyperthreading, e.g.:

#PBS -lselect=1:ncpus=16:mpiprocs=16 -lplace=scatter:excl

then the job will use 1 node with all 16 processes running on that node. Suppose this job takes 1800 seconds to complete.

Without hyperthreading, we used 2 nodes for 1000 seconds (a total of 2000 node-seconds); with hyperthreading we used 1 node for 1800 seconds (1800 node-seconds). Thus, under these circumstances, if you were interested in getting the best wall-clock time performance for a single job, you would use two nodes without hyperthreading. However, if you were interested in minimizing resource usage, especially with multiple jobs running simultaneously, use of hyperthreading would save you 10%.

An added benefit of using fewer nodes with hyperthreading, is that when Pleiades is loaded with many jobs, asking for half as many nodes may allow your job to start running sooner, resulting with an improvement in the throughput of your jobs.

Caution: Hyperthreading does not benefit all applications. Some applications may also show improvement with some process counts but not with other process counts (e.g., a 256-process Overflow job shows benefit with hyperthreading, while a 32-process Overflow job does not). There may also be other unforeseen issues with hyperthreading. Users should test their applications with and without hyperthreading before making a choice for production runs. If your application runs more than 2 times slower with hyperthreading than without hyperthreading, then it should not be used.

Turbo Boost:

On Nehalem-EP (and also Westmere), Turbo Boost is available.

When Turbo Boost is enabled, idle cores are turned off and and power is channeled to the cores that are active, making them more efficient. The net effect is that the active cores perform above their clock speed (i.e., overclocked).

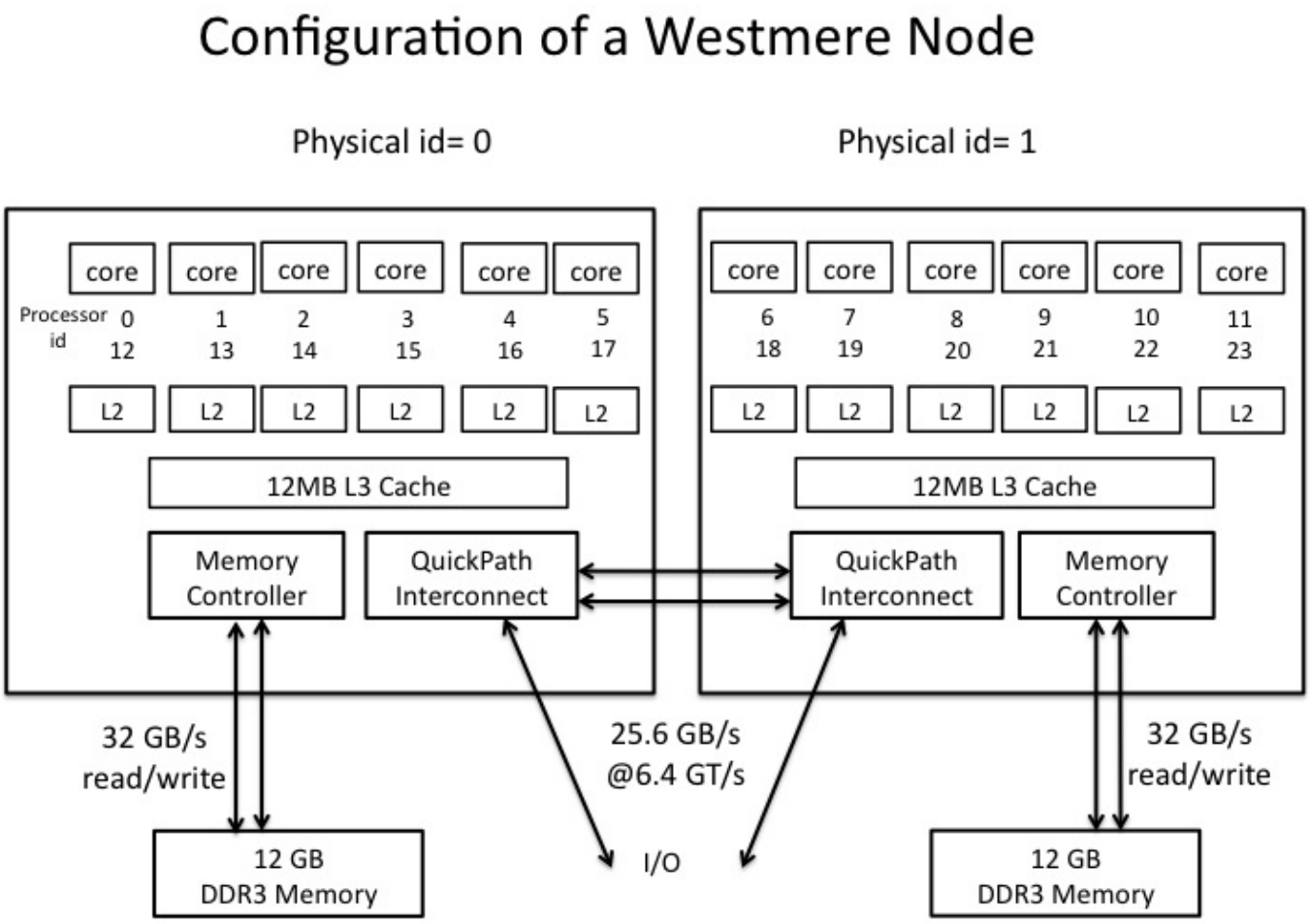
Turbo Boost mode is set up in the system BIOS. It is currently set to OFF.

Westmere Processors

DRAFT

This article is being reviewed for completeness and technical accuracy.

Configuration of a Westmere node:



Core Labeling:

Unlike Harpertown, the core labeling in Westmere is contiguous. That is, cores 0-5 are in first socket and cores 6-11 are in the second socket.

When using the SGI MPT library, the enviroment variable **MPI_DSM_DISTRIBUTE** is set to *on* by default for the Westmere nodes.

SSE4 Instruction Set:

Intel's Streaming SIMD Extensions 4.2 (SSE4.2) instruction set is included in the Westmere processors.

Since the instruction set is upward compatible, an application that is compiled with -xSSE4.1 (with Intel version 11 compiler) can run on either Harpertown or Nehalem-EP or Westmere processors. An application that is compiled with -xSSE4.2 can run ONLY on Nehalem-EP or Westmere processors.

If you wish to have a single executable that will run on any of the three Pleiades processor types with suitable optimization to be determined at run time, you can compile your application with -O3 -ipo -axSSE4.2,xSSE4.1

Hyperthreading:

Hyperthreading is available by user request on the Westmere nodes (for example, by asking for more than 12 ranks per node).

Turbo Boost:

Turbo Boost is set to *off* on the Westmere nodes.

Comparison among Harpertown, Nehalem-EP and Westmere

DRAFT

This article is being reviewed for completeness and technical accuracy.

Among the three processor types used in Pleiades, Nehalem-EP and Westmere are very similar to each other, while Harpertown is significantly different from the other two.

The main differences between Nehalem-EP and Westmere processors are:

- Both Nehalem-EP and Westmere have 24 GB of memory per node. However, there are 8 cores per Nehalem-EP node vs 12 cores per Westmere node, resulting in more memory per core for Nehalem-EP (3 GB/core) than Westmere (2 GB/core).

- The size of the L3 cache is 8 MB per quad-core for Nehalem-EP while it is 12 MB per 6-core for Westmere.

- For inter-node communication, two devices are involved: the Infiniband switches and the host channel adapter chip (HCA). For both the Nehalem-EP and Westmere racks, there are two SGI Infiniband QDR switches per half-IRU (which comprises 8 nodes). One of the switches is used for ib0 (used mainly for MPI communication), and the other for ib1 (used mainly for IO). The maximum raw data transfer rate through these switches is 40 Gigabits per second (Gbps). However, the HCA on each motherboard (one node per motherboard) is different for Nehalem-EP and Westmere. For Nehalem-EP, a 4x DDR HCA with a raw data transfer rate of 20 Gbps is used. For Westmere, a 4x QDR HCA with a rate of 40 Gbps is used. This difference results in better inter-node communication performance between the Westmere nodes than between the Nehalem-EP nodes.

  Note: The communication path between pairs of nodes vary, depending on where the nodes are relative to each other. For example:

  - two nodes on the same half-IRU: HCA to IB switch on the half-IRU to HCA.
  - two nodes on the same IRU but different half-IRUs: HCA to IB switch (of one half-IRU) to IB switch (of the other half-IRU) to HCA.

The main differences between Harpertown and Nehalem-EP/Westmere processors are:

- The processor labeling in Harpertown is not contiguous. On the contrary, the labeling in Nehalem-EP/Westmere is contiguous.

- Nehalem-EP/Westmere incorporates the SSE 4.2 SIMD instructions, which adds 7

new instructions to the SSE 4.1 set in Harpertown.

- Every two cores in Harpertown share a common L2 cache, while every core in Nehalem-EP/Westmere has its own private L2 cache. In addition, there is a L3 cache shared by the four cores in each socket of Nehalem-EP (or by the 6 cores in each socket of Westmere), while there is none for Harpertown.

- The Nehalem-EP based nodes have 3 GB/core (i.e., 24 GB/node) of memory as compared to 1 GB/core (i.e., 8 GB/node) in most of the Harpertown-based nodes in Pleiades.

  The Westmere based nodes have 2 GB/core (i.e., 24 GB/node) of memory as compared to 1 GB/core (i.e., 8 GB/node) in most of the Harpertown-based nodes in Pleiades.

- Nehalem-EP/Westmere, with a higher ratio of memory bandwidth to processor speed, is a better balanced system than the Harpertown.

  The key features which enable this improvement are the Intel QuickPath Interconnect, which provides communication with the other processor on the same node, and an integrated memory controller. Together they result in a higher aggregate bandwidth.

  In addition, each Nehalem-EP/Westmere core has its own L1 and L2 cache which helps to decrease the number of stalls in a data path. The data pre-fetch algorithm for L2 and L3 caches has been substantially reworked to achieve more effective data loads.

- Hyperthreading and TurboBoost are additional features on Nehalem-EP/Westmere, but not for Harpertown. Hyperthreading is available by user request for Nehalem-EP/Westmere. Turbo Boost is set to *off* for Nehalem-EP/Westmere.

Pleiades Home Filesystem

DRAFT

This article is being reviewed for completeness and technical accuracy.

The home file system on Pleiades (/u/username) is an SGI NEXIS 9000 filesystem. It is NFS-mounted on all of the Pleiades front-ends, bridge nodes and compute nodes.

Once a user is granted an account on Pleiades, the home directory is set up automatically during his/her first login.

Quota and Policy

Disk space quota limits are enforced on the home filesystem. By default, the soft limit is 8GB and the hard limit is 10GB. There are no inode limits on the home filesystem.

To check your quota and usage on your home filesystem, do:

```
%quota -v
Disk quotas for user username (uid xxxx):
     Filesystem  blocks   quota   limit   grace   files   quota   limit   grace
saturn-ib1-0:/mnt/home2
                7380152  8000000 40000000          190950       0       0
```

The quota policy for NAS states that if you exceed the soft quota, an email will be sent to inform you of your current usage and how much of your grace period remains. It is expected that a user will occasionally exceed their soft limit as needed, however after 14 days, users who are still over their soft limit will have their batch queue access to Pleiades disabled. If you believe that you have a long-term need for higher quota limits, you should send an email justification to support@nas.nasa.gov. This will be reviewed by the HECC Deputy Project Manager, Bill Thigpen, for approval.

The quota policy for NAS can be found here.

Backup Policy

Files on the home filesystem are backed up daily.

Pleiades Lustre Filesystems

Pleiades has several Lustre filesystems (/nobackupp[10-60]) that provide a total of about 3 PB of storage and serve thousands of cores. These filesystems are managed under Lustre software version 1.8.2.

Lustre filesystem configurations are summarized at the end of this article.

Which /nobackup should I use?

Once you are granted an account on Pleiades, you will be assigned to use one of the Lustre filesystems. You can find out which Lustre filesystem you have been assigned to by doing the following:

```
pfe1% ls -l /nobackup/your_username
lrwxrwxrwx 1 root root 19 Feb 23  2010 /nobackup/username -> /nobackupp30/username
```

In the above example, the user is assigned to /nobackupp30 and a symlink is created to point the user's default /nobackup to /nobackupp30.

TIP: Each Pleiades Lustre filesystem is shared among many users. To get good I/O performance for your applications and avoid impeding I/O operations of other users, read the articles: Lustre Basics and Lustre Best Practices.

Default Quota and Policy on /nobackup

Disk space and inodes quotas are enforced on the /nobackup filesystems. The default soft and hard limits for inodes are 75,000 and 100,000, respectively. Those for the disk space are 200GB and 400GB, respectively. To check your disk space and inodes usage and quota on your /nobackup, use the *lfs* command and type the following:

```
%lfs quota -u username /nobackup/username
Disk quotas for user username (uid xxxx):
   Filesystem  kbytes       quota   limit   grace   files   quota   limit   grace
/nobackup/username 1234  210000000 420000000    -     567   75000  100000       -
```

The NAS quota policy states that if you exceed the soft quota, an email will be sent to inform you of your current usage and how much of your grace period remains. It is expected that users will occasionally exceed their soft limit, as needed; however after 14 days, users who are still over their soft limit will have their batch queue access to Pleiades disabled.

If you anticipate having a long-term need for higher quota limits, please send a justification via email to support@nas.nasa.gov. This will be reviewed by the HECC Deputy Project Manager for approval.

For more information, see also, Quota Policy on Disk Space and Files.

NOTE: If you reach the hard limit while your job is running, the job will die prematurely without providing useful messages in the PBS output/error files. A Lustre error with code -122 in the system log file indicates that you are over your quota.

In addition, when a Lustre filesystem is full, jobs writing to it will hang. A Lustre error with code -28 in the system log file indicates that the filesystem is full. The NAS Control Room staff normally will send out emails to the top users of a filesystem asking them to clean up their files.

Important: Backup Policy

As the names suggest, these filesystems are not backed up, so any files that are removed *cannot* be restored. Essential data should be stored on Lou1-3 or onto other more permanent storage.

Configurations

In the table below, /nobackupp[10-60] have been abbreviated as p[10-60].

Pleiades Lustre Configurations

Filesystem	p10	p20	p30	p40	p50	p60
# of MDSes	1	1	1	1	1	1
# of MDTs	1	1	1	1	1	1
size of MDTs	1.1T	1.0T	1.2T	0.6T	0.6T	0.6T
# of usable inodes on MDTs	~235x10^6	~115x10^6	~110x10^6	~57x10^6	~113x10^6	~123x10^6
# of OSSes	8	8	8	8	8	8
# of OSTs	120	60	120	60	60	60
size/OST	7.2T	7.2T	3.5T	3.5T	7.2T	7.2T
Total Space	862T	431T	422T	213T	431T	431T
Default Stripe Size	4M	4M	4M	4M	4M	4M
Default Stripe Count	1	1	1	1	1	1

NOTE: The default stripe count and stripe size were changed on January 13, 2011. For directories created prior to this change, if you did not explictly set the stripe count and/or stripe size, the default values (stripe count 4 and stripe size 1MB) were used. This means that files created prior to January 13, 2011 had those old default values. After this date, directories without an explicit setting of stripe count and/or stripe size adopted the new stripe count of 1 and stripe size of 4MB. However, the old files in that directory will retain their old default values. New files that you create in these directories will adopt the new

default values.

Pleiades Front-End Usage Guidelines

DRAFT

This article is being reviewed for completeness and technical accuracy.

The front-end systems pfe[1-12] and bridge[1,2] provide an environment that allows you to get quick turnaround while performing the following:

- file editing
- compiling
- short debugging and testing session
- batch job submission to the compute systems

Bridge[1,2], with 4 times the memory on pfe[1-12] and better interconnects, can also be used for the following two functions:

1. **Post processing**

   These nodes have 64-bit versions of IDL, Matlab, and Tecplot installed and have 64 GB of memory (4 times the amount of memory on pfe[1-12]). The bridge nodes will run these applications much faster than on pfe[1-12].

2. **File transfer between Pleiades and Columbia or Lou**

   Note that both the Pleiades Lustre filesystems (/nobackupp[10-70]) and the Columbia CXFS filesystems (/nobackup1[1-h], /nobackup2[a-i]) are mounted on the bridge nodes.

   To copy files between the Pleiades Lustre and Columbia CXFS filesystems, log in to bridge[1,2] and use the *cp* command to perform the transfer. The 10 Gigabit Ethernet (GigE) connections on the two bridge nodes are faster than the 1 GigE used on pfe[1-12], therefore, file transfer out of Pleiades is improved when using the bridge nodes.

   File transfers from bridge[1,2] to Lou[1,2] will go over the 10 GigE interface by default. The commands *scp*, *bbftp*, and *bbscp* are available to do file transfers. Since *bbscp* uses almost the same syntax as *scp*, but performs faster than *scp*, we recommend using *bbscp* over *scp* in cases where you do not require the data to be encrypted when sent over the network.

   ```
   The pfe systems ([pfe1-12]) have a 1 GigE connection, which
   can be saturated by a single secure copy (scp). You will see
   bad performance whenever more than one file transfer is
   happening. Use of bridge1 and bridge2 for file transfers is
   ```

strongly recommended.

File transfers from the compute nodes to Lou must go through pfe[1-12] or bridge[1,2] first, although going through bridge[1,2] is preferred for performance consideration. See Transferring Files from the Pleiades Compute Nodes to Lou for more information.

When sending data to Lou[1-2], please keep your largest individual file size under 1 TB, as large files will keep all of the tape drives busy, preventing other file restores and backups. To prevent the filesystems on Lou[1-2] from filling up, please limit total data transfers to 1 TB and then wait an hour before continuing. This allows the tape drives to write the data to tape.

Additional restrictions apply to using these front-end systems:

1. No MPI jobs are allowed to run on pfe[1-12], bridge[1,2]

2. A job on pfe[1-12] should not use more than 8 GB. When it does, a courtesy email is sent to the owner of the job.

3. A job on bridge[1,2] should not use more than 56 GB. When it does, a courtesy email is sent to the owner of the job.

Pleiades Interconnect

DRAFT

This article is being reviewed for completeness and technical accuracy.

Topology

InfiniBand (IB) is used for inter-node communication among all of the Pleiades nodes. A key feature of InfiniBand permits remote direct memory access (RDMA) between processing nodes, allowing direct access to other nodes' memory. This allows developers and application owners to bypass the TCP/IP stack, accelerating the application performance. Two devices are involved in the interconnect: the Mellanox ConnectX host channel adapter chip (HCA) on the motherboard of each node and the Mellanox IB switches. There are two IB switches per half- IRU (which includes 8 nodes). One of the switches is involved in the ib0 fabric, which is used mainly for MPI communication. The other is involved in the ib1 fabric which is used mainly for I/O. InfiniBand uses subnet manager (SM) software to manage the InfiniBand fabric and to monitor interconnect performance and health at the fabric level.

The network topology of each IB fabric of Pleiades is a partial 11-D hypercube. In a 11-D hypercube, each switch has 11 direct connections with 11 other specific switches in the network.

The ib1 hypercube fabric is extended by a set of nine switches connected to the Lustre servers (one for the MDSes, and eight for the OSSes). The plan is for each rack to connect directly to one of the OSS switches and each group of eight racks to connect directly to the MDS switch. Currently, most of the racks are connected this way, but some remain to be connected.

Another set of nine switches on the ib1 fabric provides direct access between hyperwall visualization nodes and Pleiades nodes and Lustre servers.

Latency

The shortest communication path in a Pleiades IB fabric will be for any two nodes located in the same half-IRU of the same rack such that the communication only needs to go through 1 switch. For a fully populated 11-D hypercube, the optimum communication path between any two nodes which are not in the same half-IRU varies from going through 2 to 12 switches, depending on which racks and half-IRUs the two nodes reside. Since the Pleiades IB fabric is not a full 11-D hypercube, some connections are missing that would facilitate the optimum path between some nodes, therefore, it is possible that some communications may go over more than 12 switches.

MPI half Ping-Pong latency starts around 1000 to 1500 ns for communication going through two switches. Each additional switch adds ~100 ns (QDR) to ~150 ns (DDR) to the latency.

Bandwidth

The HCA on each node uses either 4x DDR (double data rate) links or 4x QDR (quad data rate) links. Each link is bi-directional and contains 1 send channel and 1 receive channel. For each direction, the raw data transfer rates for 4x DDR and 4x QDR are 20 Gb/s and 40 Gb/s, respectively. These links use 8b/10b encoding such that every 10 bits sent carry 8 bits of useful data. Thus, for each direction, the effective maximum bandwidth for each node is 16 Gb/s (i.e, 2 GB/s) if 4x DDR HCA is used or 32 Gb/s (i.e. 4 GB/sec) if 4x QDR HCA is used.

The IB switch which every 8 nodes in each half-IRU share through a single path also has similar effective data transfer limits per port: 16 Gb/s for 4x DDR IB switches and 32 Gb/s for 4x QDR IB switches.

The Harpertown and Nehalem-EP nodes use 4x-DDR HCAs while the Westmere nodes use 4x-QDR HCAs. The Harpertown racks use 4x DDR IB switches while the Nehalem-EP and Westmere racks use 4x QDR switches.

These limits also apply to each OSS in the Lustre filesystem. For /nobackupp20 and /nobackupp50, QDR switches are used to connect to the IB fabric and DDR switches are used to connect to the DDNs of the hard disks. For /nobackupp[10,30,40,60], DDR switches are used to connect to both IB fabric and to the DDNs. With DDR switches, the theoretical bandwidth of each OSS is 2 GB/s for each direction. With 8 OSSes per Lustre filesystem, the theoretical peak aggregate bandwidth for each filesystem for each direction would be 16 GB/s. This bandwidth however is reduced to 10 GB/s due to bandwidth that the DDNs can provide. The best benchmark performance obtained for each Pleiades Lustre filesystem is 8 - 10 GB/sec (all read or all write).

The actual I/O bandwidth a user's application experiences is far less than the theoretical peak or even the benchmark data due to factors such as the I/O pattern the application is doing (for example, serial or parallel; for parallel, if the I/O requests are from nodes of different half-IRUs), the number of stripe count used (this affects the maximum aggregate bandwidth provided by the OSSs), how busy the Lustre is handling requests from many users, if there are bad links in the network, etc.

Follow the tips listed in Lustre Best Practices if you are not getting good performances out of the Lustre filesystem.